1. A method for balancing a loading of a storage device attached to multiple computing systems during transfer of a requested data object to or from said data storage device, comprising the steps of:

acquiring a listing of locations of all segments of the requested data object including all copies of said segments of the requested data object;

evaluating the loading of the storage devices attached to the multiple computing systems containing all copies of all segments of a requested data object;

selecting storage devices containing copies of each segment of the data object having a least loading and is less than a maximum loading for said storage devices;

if the loading of the storage devices is greater than the maximum loading for said storage devices, generating a sub-segment list for any segment of the requested data object residing on said storage device would have a loading greater than the maximum loading if said segment of the requested data object were transferred, determining a load margin for all available storage devices, assigning locations for each sub-segment on each available storage device having the least loading, and assigning file names to each sub-segment;

selecting said each available storage device containing each segment and sub-segment; and

transferring those segments of said requested data object to a requesting computer system.

2. The method of claim 1 wherein calculating the new sub-segment size list comprises the steps of:

determining a number of all available storage devices that may retain a plurality of sub-segments of said sub-segment of the requested data object;

determining a maximum data object transfer load for the available storage devices;

assigning a minimum sub-segment size which is the smallest amount of data to be contained within one sub-segment of the segment;

calculating a first sub-segment size as a first function of a number of the storage devices, the current digital data transfer load, the maximum digital data transfer load, and the minimum sub-segment size;

assigning a last sub-segment size as the minimum sub-segment size;

calculating all remaining sub-segment sizes as a second function of the number of the storage devices, the current digital data transfer load, the maximum data object transfer load, and the minimum sub-segment size; and

partitioning said segment into sub-segments whereby the first sub-segment of the segment is of the first sub-segment size, the last sub-segment of the

segment is of the last sub-segment size, and all the remaining sub-segments of the segment is of the remaining sub-segment sized.

3. The method of claim 2 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N}_{\mathbf{d}} + \left(\mathbf{M}_{\mathbf{l}} / \mathbf{M}_{\mathbf{l}} - \mathbf{C}_{\mathbf{l}} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load, and

C_l is the current data object transfer load.

4. The method of claim 2 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize}_{min}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right)$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load, and

C_l is the current data object transfer load.

5. The method of claim 2 wherein calculating the new sub-segment size list further comprises the step of:

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

6. The method of claim 5 wherein the first function is further dependent upon the file interactivity factor.

7. The method of claim 6 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{I}$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

I is the file interactivity factor.

8. The method of claim 5 wherein the second function is further dependent upon the file interactivity factor.

9. The method of claim 8 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + I$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

I is the file Inter activity factor.

10. The method of claim 2 wherein calculating the new sub-segment size list further comprises the step of:

determining a file usage factor describing a number of requests for said data object for a period of time.

11. The method of claim 10 wherein the first function is further dependent upon the file usage factor.

12. The method of claim 11 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N}_{\mathbf{d}} + \left(\mathbf{M}_{\mathbf{l}} / \mathbf{M}_{\mathbf{l}} - \mathbf{C}_{\mathbf{l}} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load, and

H is the file usage factor.

13. The method of claim 9 wherein the second function is further dependent upon the file usage factor.

14. The method of claim 13 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize}_{\mathbf{min}}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + H$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

H is the file usage factor.

15. The method of claim 2 wherein calculating the new sub-segment size list further comprises the steps of:

determining a file usage factor describing a number of requests for said data object for a period of time; and

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

16. The method of claim 15 wherein the first function is further dependent upon the file usage factor and the file interactivity factor.

17. The method of claim 16 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + H + I$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load,

H is the file usage factor, and

I is the file Inter activity factor.

18. The method of claim 15 wherein the second function is further dependent upon the file usage factor and the file interactivity factor.

19. The method of claim 18 wherein the second function to determine the remaining sub-segment sizes is:

$$SubSegn = \max(SubSegSize_{min}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load, and

H is the file usage factor.

20. The method of claim 1 further comprising the steps of:

determining the presence of all segments and sub-segments of the requested data object;

if there are missing segments and sub-segments of the requested data object, assigning each of those missing segments and sub-segments a file identification and file location, such that those missing segments and sub-

segments are assigned to data storage devices having the least loading; and

retrieving those missing segments and sub-segments from a back-up storage device.

21. The method of claim 1 wherein selecting the storage devices containing copies of the segments of the requested data object and having the least loading comprises the steps of:

setting a current segment indicator to indicate which of the segments of the data object is to be transferred next;

setting a current storage device indicator to specify a primary location of the segment to be transferred next;

if the transfer of said segment causes the loading of the storage device containing said segment to be exceeded, incrementing the current storage device indicator to a next location of the segment to be transferred; and

repeated executing step c) until said loading is not exceeded.

22. The method of claim 1 wherein transferring of the segments of the data object comprises the actions of reading said segments from the data storage device, writing said segments to the data storage device, and copying said segments from a said data storage device to an alternate data storage device, whereby said

loading of the data storage device is allocated between the reading, writing, and copying of the segments to prevent interference with said reading of the segments.

23. The method of claim 1 wherein the requested data object is a video data file to be streamed isochronously to the requesting computer system.

24. A data object service system in communication with a plurality of computing systems to provide at least one data object of plurality of data object to at least one of the plurality of computing system, comprising:

a plurality of data object storage devices in communication with each other and with any of the plurality of computing systems; and

a load balancing apparatus in communication with the plurality of data object storage devices to balance a loading of said data object storage devices during transfer of said data objects,

whereby said load balancing apparatus comprises:

a load evaluator to assess the loading of the data object storage devices containing segments of said data objects,

a storage device selector to create a selection list to indicate selection of those data object storage devices containing copies of each segment of the requested data object having the least loading, and

a coping initiator to initiate a copying and further segmenting of a segment of the data object to an alternate storage devices having low loading if all storage devices containing said segment have a loading greater than a maximum loading, whereby said copying initiator generates a sub-segment list for any segment of the requested data object residing on said storage device that would have a loading greater than the maximum loading if said segment of the requested data object were transferred, determines a load margin for all available storage devices, assigns locations for each sub-segment on each available storage device having the least loading, and assigns file names to each sub-segment;

25. The system of claim 24 wherein calculating the new sub-segment size list comprises the steps of:

determining a number of all available storage devices that may retain a plurality of sub-segments of said sub-segment of the requested data object;

determining a maximum data object transfer load for the available storage devices;

assigning a minimum sub-segment size which is the smallest amount of data to be contained within one sub-segment of the segment;

calculating a first sub-segment size as a first function of a number of the storage devices, the current digital data transfer load, the maximum digital data transfer load, and the minimum sub-segment size;

assigning a last sub-segment size as the minimum sub-segment size;

calculating all remaining sub-segment sizes as a second function of the number of the storage devices, the current digital data transfer load, the maximum data object transfer load, and the minimum sub-segment size; and

partitioning said segment into sub-segments whereby the first sub-segment of the segment is of the first sub-segment size, the last sub-segment of the segment is of the last sub-segment size, and all the remaining sub-segments of the segment is of the remaining sub-segment sized.

26. The system of claim 25 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, V/f)$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load, and

$\mathbf{C_l}$ is the current data object transfer load.

27. The system of claim 25 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

(b) $\mathbf{M_l}$ is the maximum data object transfer load, and

(c) $\mathbf{C_l}$ is the current data object transfer load.

28. The system of claim 25 wherein calculating the new sub-segment size list further comprises the step of:

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

29. The system of claim 28 wherein the first function is further dependent upon the file interactivity factor.

30. The system of claim 29 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{I}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

(b) $\mathbf{M_l}$ is the maximum data object transfer load,

(c) C_l is the current data object transfer load, and

(d) **I** is the file interactivity factor.

31. The system of claim 28 wherein the second function is further dependent upon the file interactivity factor.

32. The system of claim 31 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize}_{min}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + I$$

where

(a) N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

(b) M_l is the maximum data object transfer load,

(c) C_l is the current data object transfer load, and

(d) **I** is the file Inter activity factor.

33. The system of claim 25 wherein calculating the new sub-segment size list further comprises the step of:

determining a file usage factor describing a number of requests for said data object for a period of time.

34. The system of claim 33 wherein the first function is further dependent upon the file usage factor.

35. The system of claim 34 wherein the first function to determine the first sub-segment size is:

$$SubSeg1 = \min(SubSegSize_{min}, V/f)$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

(b) $\mathbf{M_l}$ is the maximum data object transfer load,

(c) $\mathbf{C_l}$ is the current data object transfer load, and

(d) **H** is the file usage factor.

36. The system of claim 33 wherein the second function is further dependent upon the file usage factor.

37. The system of claim 36 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

(b) M_l is the maximum data object transfer load,

(c) C_l is the current data object transfer load, and

(d) **H** is the file usage factor.

38. The system of claim 25 wherein calculating the new sub-segment size list further comprises the steps of:

determining a file usage factor describing a number of requests for said data object for a period of time; and

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

39. The system of claim 38 wherein the first function is further dependent upon the file usage factor and the file interactivity factor.

40. The system of claim 39 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \min(\mathbf{SubSegSize}_{min}, V/f)$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H} + \mathbf{I}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

(b) $\mathbf{M_l}$ is the maximum data object transfer load,

(c) $\mathbf{C_l}$ is the current data object transfer load,

(d) **H** is the file usage factor, and

(e) **I** is the file Inter activity factor.

41. The system of claim 38 wherein the second function is further dependent upon the file usage factor and the file interactivity factor.

42. The system of claim 41 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, V/f)$$

where

(a) **SubSegn** is the a sub-segment size for one sub-segment of the remaining sub-segments,

(b) **max** is the maximum function of two variables,

(c) **V** is a total size of a segment of the data object, and

(d) **f** is determined by the formula:

(i) $$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

(ii) where

(iii) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

(iv) M_l is the maximum data object transfer load,

(v) C_l is the current data object transfer load, and

(vi) **H** is the file usage factor.

43. The system of claim 24 wherein the load balancing apparatus generates a listing of all copies of all segments of the requested data objects.

44. The system of claim 24 wherein a data storage device having a copied segment of the requested data object is selected to transfer said data object to a requesting computing system.

45. The system of claim 24 wherein the load balancing apparatus comprises:

a presence determining device to determine the presence of all segments and sub-segments of the requested data object; and

a segment retrieving device which, if there are missing segments and sub-segments and sub-segments of the requested data object, said segment retrieving device assigns each of those missing segments and sub segments a file identification and file location, such that those missing segments are assigned to data storage devices having the least loading, and then said segment retrieving device retrieves those missing segments from a back-up storage device.

46. The system of claim 24 wherein the load balancing apparatus further comprises:

a segment indicator that is set to indicate which of the segments of the data object to be transferred next; and

a current storage device indicator to specify initially a primary location of the segment identified by the segment indicatory;

whereby the load balancing apparatus

determines if the loading of the data storage device indicated by the current storage device indicator exceeds the maximum loading with transfer of the segment indicated by the segment indicator,

if said loading exceeds the maximum loading, the current storage device indicator is set to a next location of the segment indicated by the segment indicator, and

repeated executes steps a) and b) until said loading is not exceeded.

47. The system of claim 24 wherein transferring of the segments of the data object comprises the actions of reading said segments from the data storage device, writing said segments to the data storage device, and copying said segments from a said data storage device to an alternate data storage device, whereby said loading of the data storage device is allocated between the reading, writing, and copying of the segments to prevent interference with said reading of the segments.

48. The system of claim 24 wherein the requested data object is a video data file to be streamed isochronously to the requesting computer system.

49. An apparatus for balancing a loading of a storage device attached to multiple computing systems comprising:

means for acquiring a listing of locations of all segments of a requested data object including all copies of said segments and sub-segments of the requested data object;

means for evaluating the loading of the storage devices attached to the multiple computing systems containing all copies of all segments of a requested data object;

means for selecting storage devices containing copies of each segment of the data object having a least loading, which is less than a maximum loading for said storage devices;

means for generating a sub-segment list for any segment of the requested data object residing on said storage device would have a loading greater than the maximum loading if said segment of the requested data object were transferred,

means for determining a load margin for all available storage devices;

means for assigning locations for each sub-segment on each available storage device having the least loading;

means for assigning file names to each sub-segment,

whereby said means for generating a sub-segment list, said means for determining a load margin, said means for assigning locations for each sub-segment, and said means for assigning file names function if the loading of the storage devices is greater than the maximum loading for said storage devices;

means for selecting said each available storage device containing each sub-segment; and

means for transferring those segments of said requested data object to a requesting computer system.

50. The apparatus of claim 49 wherein calculating the new sub-segment size list comprises the steps of:

determining a number of all available storage devices that may retain a plurality of sub-segments of said sub-segment of the requested data object;

determining a maximum data object transfer load for the available storage devices;

assigning a minimum sub-segment size which is the smallest amount of data to be contained within one sub-segment of the segment;

calculating a first sub-segment size as a first function of a number of the storage devices, the current digital data transfer load, the maximum digital data transfer load, and the minimum sub-segment size;

assigning a last sub-segment size as the minimum sub-segment size;

calculating all remaining sub-segment sizes as a second function of the number of the storage devices, the current digital data transfer load, the maximum data object transfer load, and the minimum sub-segment size; and

partitioning said segment into sub-segments whereby the first sub-segment of the segment is of the first sub-segment size, the last sub-segment of the segment is of the last sub-segment size, and all the remaining sub-segments of the segment is of the remaining sub-segment sized.

51. The apparatus of claim 50 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load, and

$\mathbf{C_l}$ is the current data object transfer load.

52. The apparatus of claim 51 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load, and

$\mathbf{C_l}$ is the current data object transfer load.

53. The apparatus of claim 50 wherein calculating the new sub-segment size list further comprises the step of:

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

54. The apparatus of claim 53 wherein the first function is further dependent upon the file interactivity factor.

55. The apparatus of claim 54 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{I}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load, and

I is the file interactivity factor.

56. The apparatus of claim 53 wherein the second function is further dependent upon the file interactivity factor.

57. The apparatus of claim 56 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize}_{min}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + I$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

I is the file Inter activity factor.

58. The apparatus of claim 50 wherein calculating the new sub-segment size list further comprises the step of:

determining a file usage factor describing a number of requests for said data object for a period of time.

59. The apparatus of claim 58 wherein the first function is further dependent upon the file usage factor.

60. The apparatus of claim 59 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N}_{\mathbf{d}} + \left(\mathbf{M}_{\mathbf{l}} / \mathbf{M}_{\mathbf{l}} - \mathbf{C}_{\mathbf{l}} \right) + \mathbf{H}$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

H is the file usage factor.

61. The apparatus of claim 58 wherein the second function is further dependent upon the file usage factor.

62. The apparatus of claim 61 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + H$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

H is the file usage factor.

63. The apparatus of claim 50 wherein calculating the new sub-segment size list further comprises the steps of:

determining a file usage factor describing a number of requests for said data object for a period of time; and

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

64. The apparatus of claim 63 wherein the first function is further dependent upon the file usage factor and the file interactivity factor.

65. The apparatus of claim 64 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H} + \mathbf{I}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load,

H is the file usage factor, and

I is the file Inter activity factor.

66. The apparatus of claim 63 wherein the second function is further dependent upon the file usage factor and the file interactivity factor.

67. The apparatus of claim 66 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize}_{min}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + H$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

H is the file usage factor.

68. The apparatus of claim 49 further comprising:

means for determining the presence of all segments and sub-segments of the requested data object;

means for assigning each of those missing segments a file identification and file location, such that those missing segments are assigned to data storage devices having the least loading, if there are missing segments and sub-segments of the requested data object; and

means for retrieving those missing segments from a back-up storage device.

69. The apparatus of claim 68 wherein the means for selecting the storage devices containing copies of the segments and sub-segments of the requested data object and having the least loading comprises:

means for setting a current segment indicator to indicate which of the segments of the data object is to be transferred next;

means for setting a current storage device indicator to specify a primary location of the segment to be transferred next;

means for incrementing the current storage device indicator to a next location of the segment to be transferred, if the transfer of said segment causes the loading of the storage device containing said segment to be exceeded; and

means for repeated executing step c) until said loading is not exceeded.

70. The apparatus of claim 68 wherein transferring of the segments of the data object comprises the actions of reading said segments from the data storage device, writing said segments to the data storage device, and copying said segments from a said data storage device to an alternate data storage device, whereby said loading of the data storage device is allocated between the reading, writing, and copying of the segments to prevent interference with said reading of the segments.

71. The apparatus of claim 68 wherein the requested data object is a video data file to be streamed isochronously to the requesting computer system.

72. A medium for retaining a computer program which, when executed on a computing system, balances a loading of storage devices attached to multiple computing systems, whereby said program executes the steps of:

acquiring a listing of locations of all segments of a requested data object including all copies of said segments and sub-segments of the requested data object;

evaluating the loading of the storage devices attached to the multiple computing systems containing all copies of all segments of a requested data object;

selecting storage devices containing copies of each segment of the data object having a least loading, which is less than a maximum loading for said storage devices;

if the loading of the storage devices is greater than the maximum loading for said storage devices, generating a sub-segment list for any segment of the requested data object residing on said storage device would have a loading greater than the maximum loading if said segment of the requested data object were transferred, determining a load margin for all available storage devices, assigning locations for each sub-segment on each available storage device having the least loading, and assigning file names to each sub-segment;

selecting said each available storage device containing each sub-segment; and

transferring those segments of said requested data object to a requesting computer system.

73. The medium of claim 72 wherein calculating the new sub-segment size list comprises the steps of:

determining a number of all available storage devices that may retain a plurality of sub-segments of said sub-segment of the requested data object;

determining a maximum data object transfer load for the available storage devices;

assigning a minimum sub-segment size which is the smallest amount of data to be contained within one sub-segment of the segment;

calculating a first sub-segment size as a first function of a number of the storage devices, the current digital data transfer load, the maximum digital data transfer load, and the minimum sub-segment size;

assigning a last sub-segment size as the minimum sub-segment size;

calculating all remaining sub-segment sizes as a second function of the number of the storage devices, the current digital data transfer load, the maximum data object transfer load, and the minimum sub-segment size; and

partitioning said segment into sub-segments whereby the first sub-segment of the segment is of the first sub-segment size, the last sub-segment of the segment is of the last sub-segment size, and all the remaining sub-segments of the segment is of the remaining sub-segment sized.

74. The medium of claim 73 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load, and

$\mathbf{C_l}$ is the current data object transfer load.

75. The medium of claim 73 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load, and

$\mathbf{C_l}$ is the current data object transfer load.

76. The medium of claim 73 wherein calculating the new sub-segment size list further comprises the step of:

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

77. The medium of claim 76 wherein the first function is further dependent upon the file interactivity factor.

78. The medium of claim 77 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{I}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load, and

I is the file interactivity factor.

79. The medium of claim 73 wherein the second function is further dependent upon the file interactivity factor.

80. The medium of claim 79 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, V/f)$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$f = N_d + \left(M_l / M_l - C_l \right) + l$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

I is the file Inter activity factor.

81. The medium of claim 73 wherein calculating the new sub-segment size list further comprises the step of:

determining a file usage factor describing a number of requests for said data object for a period of time.

82. The medium of claim 81 wherein the first function is further dependent upon the file usage factor.

83. The medium of claim 82 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N}_d + \left(\mathbf{M}_I \Big/ \mathbf{M}_I - \mathbf{C}_I \right) + \mathbf{H}$$

where

N_d is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

H is the file usage factor.

84. The medium of claim 81 wherein the second function is further dependent upon the file usage factor.

85. The medium of claim 84 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load, and

H is the file usage factor.

86. The medium of claim 73 wherein calculating the new sub-segment size list further comprises the steps of:

determining a file usage factor describing a number of requests for said data object for a period of time; and

determining a file interactivity factor describing a number of jumps by the second computing system within the data object.

87. The medium of claim 86 wherein the first function is further dependent upon the file usage factor and the file interactivity factor.

88. The medium of claim 87 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l}\right) + \mathbf{H} + \mathbf{I}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

$\mathbf{M_l}$ is the maximum data object transfer load,

$\mathbf{C_l}$ is the current data object transfer load,

H is the file usage factor, and

I is the file Inter activity factor.

89. The medium of claim 86 wherein the second function is further dependent upon the file usage factor and the file interactivity factor.

90. The medium of claim 89 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of a segment of the data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_I} / \mathbf{M_I} - \mathbf{C_I} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the segments of the data object,

M_l is the maximum data object transfer load,

C_l is the current data object transfer load, and

H is the file usage factor.

91. The medium of claim 73 further comprising the steps of:

determining the presence of all segments and sub-segments of the requested data object;

if there are missing segments and sub-segments of the requested data object, assigning each of those missing segments a file identification and file location, such that those missing segments are assigned to data storage devices having the least loading; and

retrieving those missing segments from a back-up storage device.

92. The medium of claim 91 wherein selecting the storage devices containing copies of the segments and sub-segments of the requested data object and having the least loading comprises the steps of:

setting a current segment indicator to indicate which of the segments of the data object is to be transferred next;

setting a current storage device indicator to specify a primary location of the segment to be transferred next;

if the transfer of said segment causes the loading of the storage device containing said segment to be exceeded, incrementing the current storage device indicator to a next location of the segment to be transferred; and

repeated executing step c) until said loading is not exceeded.

93. The medium of claim 73 wherein transferring of the segments of the data object comprises the actions of reading said segments from the data storage device, writing said segments to the data storage device, and copying said segments from a said data storage device to an alternate data storage device, whereby said loading of the data storage device is allocated between the reading, writing, and copying of the segments to prevent interference with said reading of the segments.

94. The medium of claim 55 wherein the requested data object is a video data file to be streamed isochronously to the requesting computer system.

95. A method for balancing a loading of a storage device containing video data objects attached to multiple computing systems comprising the steps of:

acquiring a listing of locations of all segments of a requested video data object including all copies of said segments of the requested video data object;

evaluating the loading of the storage devices attached to the multiple computing systems containing all copies of all segments of a requested video data object;

selecting storage devices containing copies of each segment of the video data object having a least loading, which is less than a maximum loading for said storage devices;

if the loading of the storage devices is greater than the maximum loading for said storage devices, generating a sub-segment list for any segment of the requested video data object residing on said storage device would have a loading greater than the maximum loading if said segment of the requested video data object were transferred, determining a load margin for all available storage devices, assigning locations for each sub-segment on each available storage device having the least loading, and assigning file names to each sub-segment;

selecting said each available storage device containing each sub-segment; and

transferring those segments of said requested video data object to a requesting computer system.

96. The method of claim 95 wherein calculating the new sub-segment size list comprises the steps of:

determining a number of all available storage devices that may retain a plurality of sub-segments of said sub-segment of the requested video data object;

determining a maximum video data object transfer load for the available storage devices;

assigning a minimum sub-segment size which is the smallest amount of data to be contained within one sub-segment of the segment;

calculating a first sub-segment size as a first function of a number of the storage devices, the current digital data transfer load, the maximum digital data transfer load, and the minimum sub-segment size;

assigning a last sub-segment size as the minimum sub-segment size;

calculating all remaining sub-segment sizes as a second function of the number of the storage devices, the current digital data transfer load, the maximum video data object transfer load, and the minimum sub-segment size; and

partitioning said segment into sub-segments whereby the first sub-segment of the segment is of the first sub-segment size, the last sub-segment of the segment is of the last sub-segment size, and all the remaining sub-segments of the segment is of the remaining sub-segment sized.

97. The method of claim 96 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, V/f)$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load, and

$\mathbf{C_l}$ is the current video data object transfer load.

98. The method of claim 96 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load, and

$\mathbf{C_l}$ is the current video data object transfer load.

99. The method of claim 96 wherein calculating the new sub-segment size list further comprises the step of:

determining a file interactivity factor describing a number of jumps by the second computing system within the video data object.

100. The method of claim 99 wherein the first function is further dependent upon the file interactivity factor.

101. The method of claim 100 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V}/\mathbf{f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{I}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load,

$\mathbf{C_l}$ is the current video data object transfer load, and

I is the file interactivity factor.

102. The method of claim 99 wherein the second function is further dependent upon the file interactivity factor.

103. The method of claim 102 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_I} / \mathbf{M_I} - \mathbf{C_I} \right) + \mathbf{I}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

M_l is the maximum video data object transfer load,

C_l is the current video data object transfer load, and

I is the file Inter activity factor.

104. The method of claim 96 wherein calculating the new sub-segment size list further comprises the step of:

determining a file usage factor describing a number of requests for said video data object for a period of time.

105. The method of claim 104 wherein the first function is further dependent upon the file usage factor.

106. The method of claim 105 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load,

$\mathbf{C_l}$ is the current video data object transfer load, and

H is the file usage factor.

107. The method of claim 104 wherein the second function is further dependent upon the file usage factor.

108. The method of claim 107 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load,

$\mathbf{C_l}$ is the current video data object transfer load, and

H is the file usage factor.

109. The method of claim 96 wherein calculating the new sub-segment size list further comprises the steps of:

determining a file usage factor describing a number of requests for said video data object for a period of time; and

determining a file interactivity factor describing a number of jumps by the second computing system within the video data object.

110. The method of claim 109 wherein the first function is further dependent upon the file usage factor and the file interactivity factor.

111. The method of claim 110 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N}_d + \left(\mathbf{M}_l \big/ \mathbf{M}_l - \mathbf{C}_l\right) + \mathbf{H} + \mathbf{I}$$

where

N_d is the number of storage devices available to retain the sub-segments of the video data object,

M_l is the maximum video data object transfer load,

C_l is the current video data object transfer load,

H is the file usage factor, and

I is the file Inter activity factor.

112. The method of claim 109 wherein the second function is further dependent upon the file usage factor and the file interactivity factor.

113. The method of claim 113 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load,

$\mathbf{C_l}$ is the current video data object transfer load, and

H is the file usage factor.

114. The method of claim 96 further comprising the steps of:

determining the presence of all segments of the requested video data object;

if there are missing segments of the requested video data object, assigning each of those missing segments a file identification and file location, such that those missing segments are assigned to data storage devices having the least loading; and

retrieving those missing segments from a back-up storage device.

115. The method of claim 96 wherein selecting the storage devices containing copies of the segments of the requested video data object and having the least loading comprises the steps of:

setting a current segment indicator to indicate which of the segments of the video data object is to be transferred next;

setting a current storage device indicator to specify a primary location of the segment to be transferred next;

if the transfer of said segment causes the loading of the storage device containing said segment to be exceeded, incrementing the current storage device indicator to a next location of the segment to be transferred; and

repeated executing step c) until said loading is not exceeded.

116. The method of claim 96 wherein transferring of the segments of the video data object comprises the actions of reading said segments from the data storage device, writing said segments to the data storage device, and copying said segments from a said data storage device to an alternate data storage device, whereby said loading of the data storage device is allocated between the reading, writing, and copying of the segments to prevent interference with said reading of the segments.

117. The method of claim 96 wherein the requested video data object is streamed isochronously to the requesting computer system.

118. A video data object service system in communication with a plurality of computing systems to provide at least one video data object of plurality of video data object to at least one of the plurality of computing system, comprising:

a plurality of video data object storage devices in communication with each other and with any of the plurality of computing systems; and

a load balancing apparatus in communication with the plurality of video data object storage devices to balance a loading of said video data object storage devices during transfer of said video data objects,

whereby said load balancing apparatus comprises:

a load evaluator to assess the loading of the video data object storage devices containing segments of said video data objects,

a storage device selector to create a selection list to indicate selection of those video data object storage devices containing copies of each segment of the requested video data object having the least loading, and

a coping initiator to initiate a copying of a segment of the video data object to an alternate storage device having low loading if all storage devices containing said segment have a loading greater than a maximum loading.

a coping initiator to initiate a copying and further segmenting of a segment of the video data object to an alternate storage devices having low loading if all storage devices containing said segment have a loading greater than a maximum loading, whereby said copying initiator generates a sub-segment list for any segment of the requested video data object residing on said storage device that would have a loading greater than the maximum loading if said segment of the requested video data object were transferred, determines a load margin for all available storage devices, assigns locations for each sub-segment on each available storage device having the least loading, and assigns file names to each sub-segment;

119. The system of claim 118 wherein calculating the new sub-segment size list comprises the steps of:

determining a number of all available storage devices that may retain a plurality of sub-segments of said sub-segment of the requested video data object;

determining a maximum video data object transfer load for the available storage devices;

assigning a minimum sub-segment size which is the smallest amount of data to be contained within one sub-segment of the segment;

calculating a first sub-segment size as a first function of a number of the storage devices, the current digital data transfer load, the maximum digital data transfer load, and the minimum sub-segment size;

assigning a last sub-segment size as the minimum sub-segment size;

calculating all remaining sub-segment sizes as a second function of the number of the storage devices, the current digital data transfer load, the maximum video data object transfer load, and the minimum sub-segment size; and

partitioning said segment into sub-segments whereby the first sub-segment of the segment is of the first sub-segment size, the last sub-segment of the segment is of the last sub-segment size, and all the remaining sub-segments of the segment is of the remaining sub-segment sized.

120. The system of claim 119 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

$\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

$\mathbf{M_l}$ is the maximum video data object transfer load, and

$\mathbf{C_l}$ is the current video data object transfer load.

121. The system of claim 119 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right)$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

(b) $\mathbf{M_l}$ is the maximum video data object transfer load, and

(c) $\mathbf{C_l}$ is the current video data object transfer load.

122. The system of claim 119 wherein calculating the new sub-segment size list further comprises the step of:

determining a file interactivity factor describing a number of jumps by the second computing system within the video data object.

123. The system of claim 122 wherein the first function is further dependent upon the file interactivity factor.

124. The system of claim 123 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize}_{\mathbf{min}}, V/f)$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{I}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

(b) $\mathbf{M_l}$ is the maximum video data object transfer load,

(c) $\mathbf{C_l}$ is the current video data object transfer load, and

(d) **I** is the file interactivity factor.

125. The system of claim 122 wherein the second function is further dependent upon the file interactivity factor.

126. The system of claim 125 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_i} / \mathbf{M_j} - \mathbf{C_j} \right) + \mathbf{I}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

(b) M_l is the maximum video data object transfer load,

(c) C_l is the current video data object transfer load, and

(d) **I** is the file Inter activity factor.

127. The system of claim 119 wherein calculating the new sub-segment size list further comprises the step of:

determining a file usage factor describing a number of requests for said video data object for a period of time.

128. The system of claim 127 wherein the first function is further dependent upon the file usage factor.

129. The system of claim 128 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

(b) $\mathbf{M_l}$ is the maximum video data object transfer load,

(c) $\mathbf{C_l}$ is the current video data object transfer load, and

(d) **H** is the file usage factor.

130. The system of claim 127 wherein the second function is further dependent upon the file usage factor.

131. The system of claim 130 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

(b) $\mathbf{M_l}$ is the maximum video data object transfer load,

(c) $\mathbf{C_l}$ is the current video data object transfer load, and

(d) **H** is the file usage factor.

132. The system of claim 119 wherein calculating the new sub-segment size list further comprises the steps of:

determining a file usage factor describing a number of requests for said video data object for a period of time; and

determining a file interactivity factor describing a number of jumps by the second computing system within the video data object.

133. The system of claim 132 wherein the first function is further dependent upon the file usage factor and the file interactivity factor.

134. The system of claim 133 wherein the first function to determine the first sub-segment size is:

$$\mathbf{SubSeg1} = \mathbf{min}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSeg1 is the first sub-segment size,

min is the minimum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l}/\mathbf{M_l} - \mathbf{C_l}\right) + \mathbf{H} + \mathbf{I}$$

where

N_d is the number of storage devices available to retain the sub-segments of the video data object,

M_l is the maximum video data object transfer load,

C_l is the current video data object transfer load,

H is the file usage factor, and

I is the file Inter activity factor.

135. The system of claim 132 wherein the second function is further dependent upon the file usage factor and the file interactivity factor.

136. The system of claim 135 wherein the second function to determine the remaining sub-segment sizes is:

$$\mathbf{SubSegn} = \mathbf{max}(\mathbf{SubSegSize_{min}}, \mathbf{V/f})$$

where

SubSegn is the a sub-segment size for one sub-segment of the remaining sub-segments,

max is the maximum function of two variables,

V is a total size of the video data object, and

f is determined by the formula:

$$\mathbf{f} = \mathbf{N_d} + \left(\mathbf{M_l} / \mathbf{M_l} - \mathbf{C_l} \right) + \mathbf{H}$$

where

(a) $\mathbf{N_d}$ is the number of storage devices available to retain the sub-segments of the video data object,

(b) $\mathbf{M_l}$ is the maximum video data object transfer load,

(c) $\mathbf{C_l}$ is the current video data object transfer load, and

(d) **H** is the file usage factor.

137. The system of claim 100 wherein the load balancing apparatus generates a listing of all copies of all segments of the requested video data objects.

138. The system of claim 100 wherein a data storage device having a copied segment of the requested video data object is selected to transfer said video data object to a requesting computing system.

139. The system of claim 100 wherein the load balancing apparatus comprises:

a presence determining device to determine the presence of all segments of the requested video data object; and

a segment retrieving device which, if there are missing segments of the requested video data object, said segment retrieving device assigns each of those missing segments a file identification and file location, such that those missing segments are assigned to data storage devices having the least loading, and then said segment retrieving device retrieves those missing segments from a back-up storage device.

140. The system of claim 118 wherein the load balancing apparatus further comprises:

a segment indicator that is set to indicate which of the segments of the video data object to be transferred next; and

a current storage device indicator to specify initially a primary location of the segment identified by the segment indicatory;

whereby the load balancing apparatus

determines if the loading of the data storage device indicated by the current storage device indicator exceeds the maximum loading with transfer of the segment indicated by the segment indicator,

if said loading exceeds the maximum loading, the current storage device indicator is set to a next location of the segment indicated by the segment indicator, and

repeated executes steps a) and b) until said loading is not exceeded.

141. The system of claim 118 wherein transferring of the segments of the video data object comprises the actions of reading said segments from the data storage device, writing said segments to the data storage device, and copying said segments from a said data storage device to an alternate data storage device, whereby said loading of the data storage device is allocated between the reading, writing, and copying of the segments to prevent interference with said reading of the segments.

142. The system of claim 118 wherein the requested video data object is streamed isochronously to the requesting computer system.